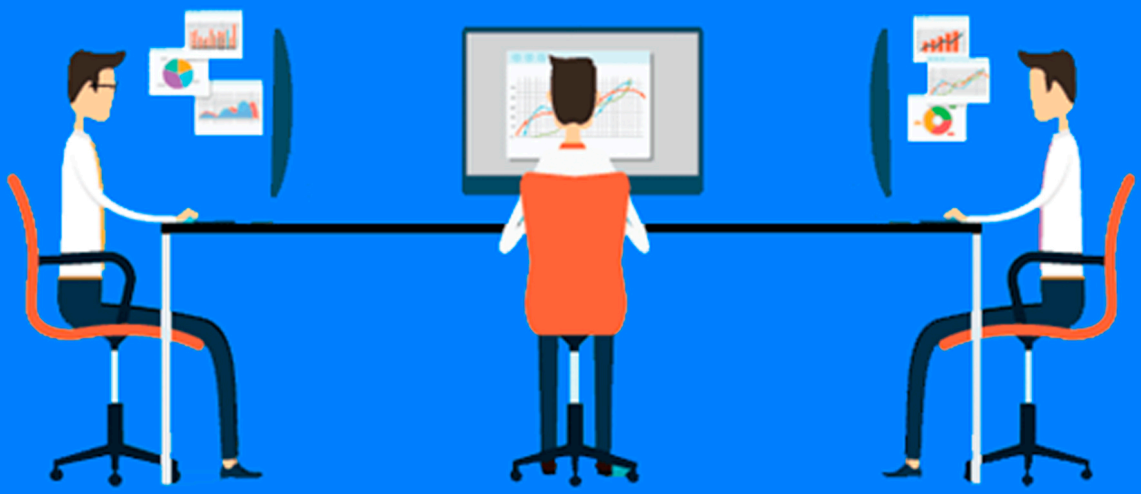
gitStarted



the zero to mastery guide
to contribute to open source
projects

ZERO TO
MASTERY

What is open source?

Open source is the original way of creating software – in an open, transparent and collaborative manner.

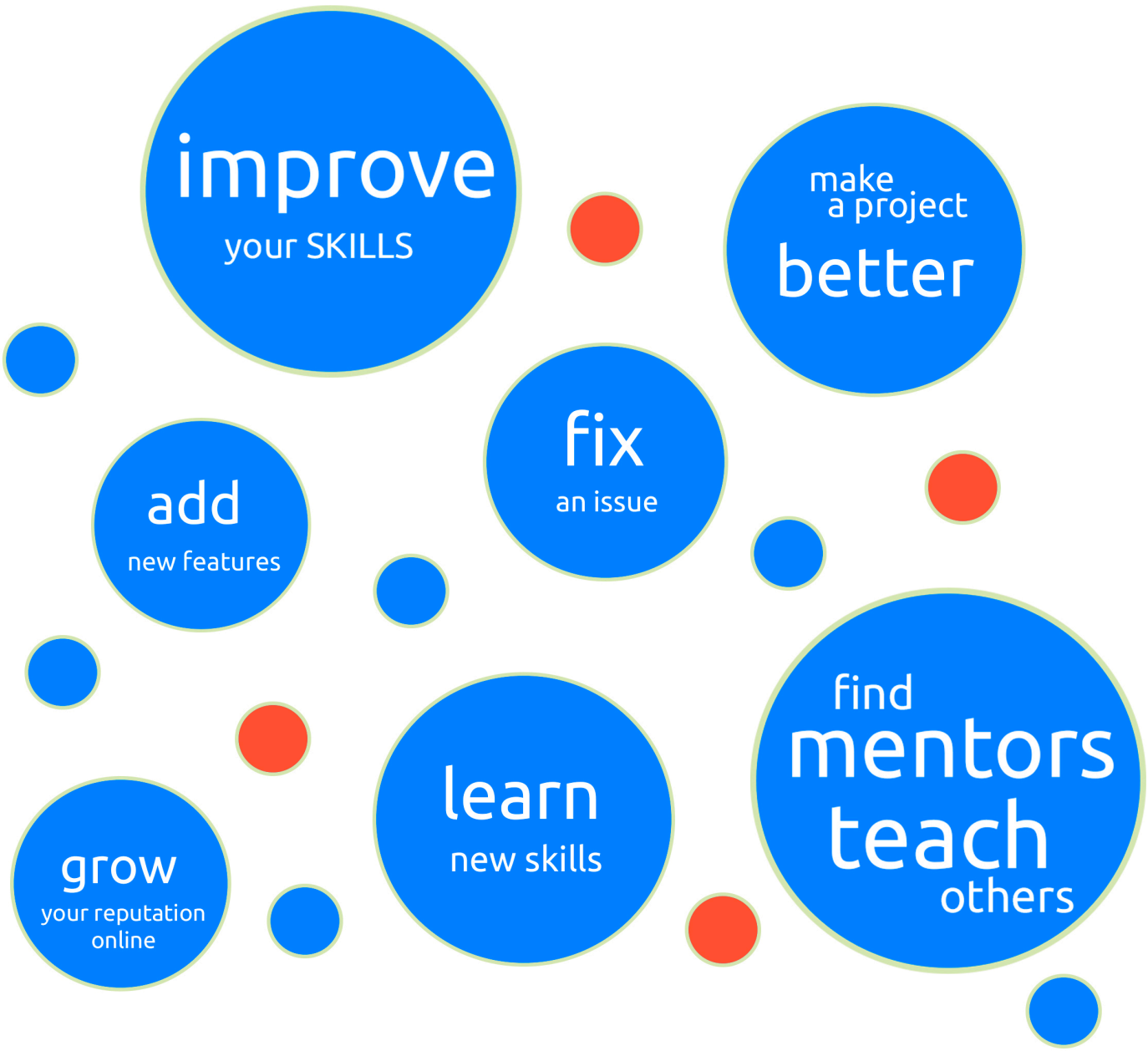
Open source means that you get the final executable for a software project, and also have access to the source code from which it was produced.

Having access to the source code means that you can better understand how the program functions and more importantly you can modify it to fix any problems you have with the software or add new features which you desire.

ZERO TO MASTERY

Why should I

contribute?



ZERO TO
MASTERY

how to contribute to open source?

*Do you **code**? You can contribute by writing your code

*Do you **design**? You can contribute by styling the project

*Do you **write**? You can improve the project documentation

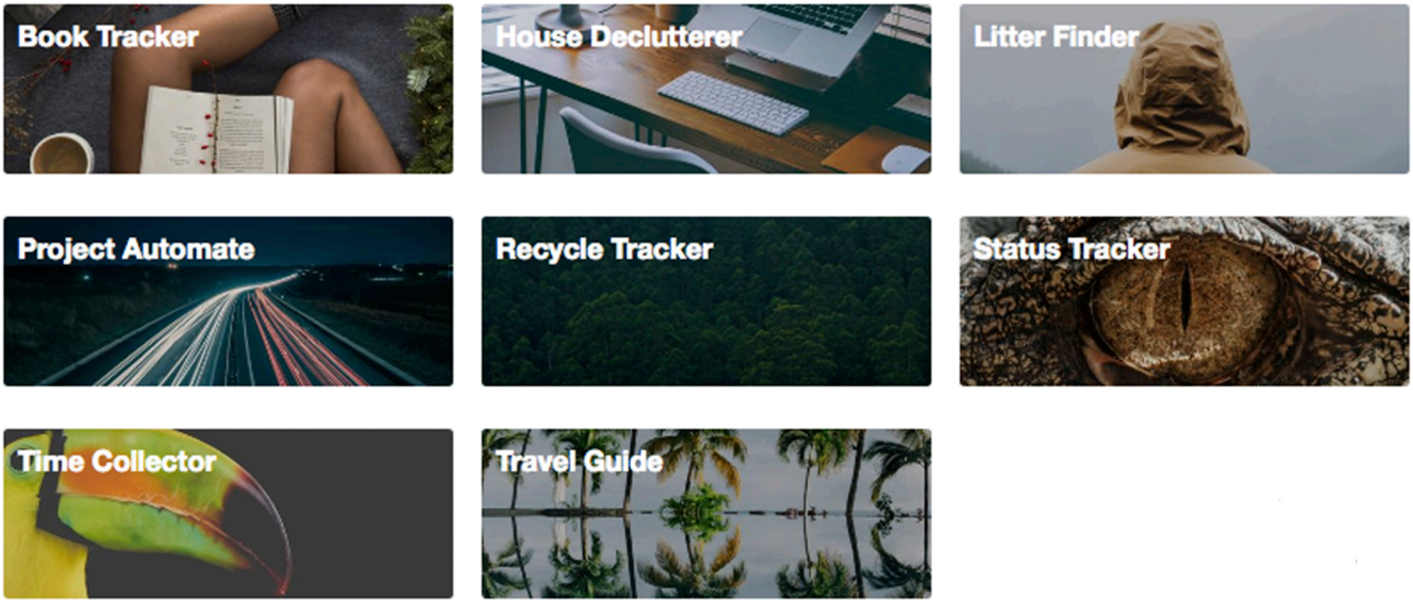
*Do you like **organizing** things? You can contribute by organizing events, deadlines, and track progress on projects

*Do you like **helping** people? You can contribute by answering their questions and guide them through

ZERO TO MASTERY

our projects

We don't care if you break things. This is a playground and we encourage failing often. Use this as a practice ground and enjoy contributing to projects you create with your fellow students.



5 steps to start

1. Read the anatomy of an open source project
2. Find out which projects you want to join
3. Sign up for the one (s) you'd like to contribute
4. Make your first pull request
5. Go join a project and start contributing!

more on: https://github.com/zero-to-mastery/start-here-guidelines

ZERO TO MASTERY

Git

It is an open source version control system (VCS). This means that when developers create something, they make constant changes to the code, releasing new versions up to and after the release.
Version control systems keep these revisions straight, storing the modifications in a central repository, allowing developers to easily collaborate. How? Download the version, make changes and upload the newest revision.

GitHub

Git is a command line tool, but the center around which all things involving Git revolve is the Hub- GitHub- where developers store their projects and network with like-minded people.
So Github is a Git repository hosting service, offering a web-based graphical interface that provides access control and other collaboration features.
Let's take you through the steps to get you started on Git and Github

first contributions

a step by step guide

Set up Git!

1) Download and install Git on your computer

2) Open Terminal and set a Git username. Type in and confirm your global username (the one for every repository)

```
$ git config --global user.name "Mona Lisa"
```

```
$ git config --global user.name
> Mona Lisa
```

3) Set an email address in Git. Type in and confirm.

```
$ git config --global user.email "email@example.com"
```

```
$ git config --global user.email
email@example.com
```

4) Add the email address to your Github account by setting the commit email address on Github, so that the commits are attributed to you and appear on your contributions graph.

ZERO TO
MASTERY

first contributions

a step by step guide

GitHub time!

1) Set up an account on Github

2) Check out the Zero to Mastery organization here and discover the different repositories.

3) You can start contributing!

Some vocabulary

Repository: also known as repo, is a location where all the files of a particular project are stored. Each project has its own repo and you can access it with a unique URL.

Forking a repo: "Forking" is when you create a new project based off of another project that already exists. If you find a project on GitHub that you'd like to contribute to, you can fork the repo, make the changes you'd like, and release the revised project as a new repo. If the original repository that you forked to create your new project gets update, you can easily add those updates to your current fork.

Pull request: you've forked a repository, made a great revision to the project and want it to be recognized by the original developer -maybe even included in the official project/repo. Pull a request.

ZERO TO MASTERY

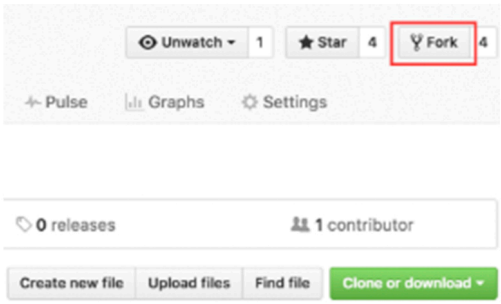
first contributions

a step by step guide

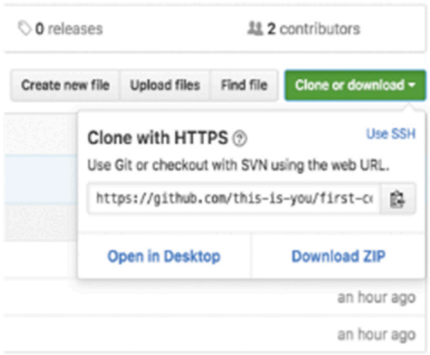
The authors of the original repository can see your work, and then choose whether or not to accept it into the official repo.

Step by step

Fork this repository clicking on the fork button. This creates a copy on your account.



Clone the repository to your computer. Click on the clone button and click the copy to clipboard icon.



Open Git terminal, type in the following git command and run it

first contributions

a step by step guide

```
git clone "url you just copied"
```

which will look something like this

```
git clone https://github.com/this-is-you/first-contributions.git
```

Change to the repo directory

```
cd first-contributions
```

Create a branch for your edits.

Branching exists to help you manage the different ideas or features in progress. When you create a branch, you're creating an environment where you can try new ideas without affecting the master branch. Therefore, experiment and commit changes, as these won't be merged until they're ready to be reviewed by other collaborator.

```
git checkout -b <add-your-name>
```

Make changes and commit them.

Whenever you add, edit or delete a file, you're making a commit, and adding them to your branch. This process keeps

first contributions

a step by step guide

track of your progress as you work on a feature branch. Commits also create a transparent history of your work that others can follow to understand what you've done and why.

For example, you open a Contributors.md file in a text editor, add your name and save it. Go to the project directory and execute `git status`, you'll see there the changes. Add those changes to the branch, with command `git add`

```
git add.
```

And commit them using `git commit`

```
git commit -m "Add <your-name> to Contributors list"
```

Push the changes to GitHub

```
git push origin <add-your-name>
```

Now, let's pull request

What we are doing here is basically propose and collaborate on changes to a repository.

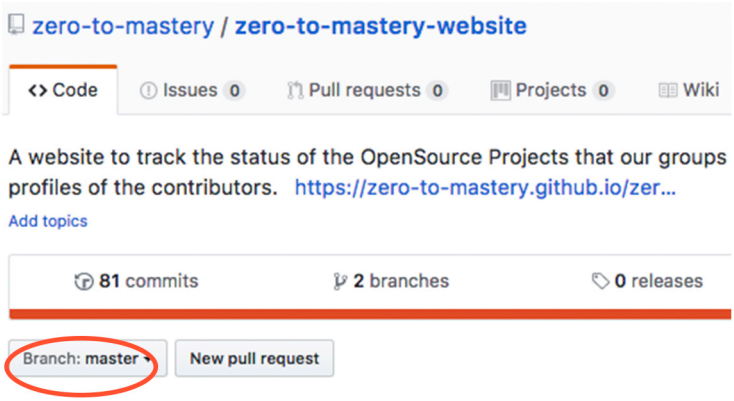
ZERO TO
MASTERY

first contributions

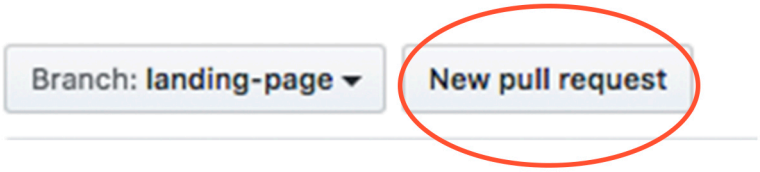
a step by step guide

1) Navigate to the main page of the repository.

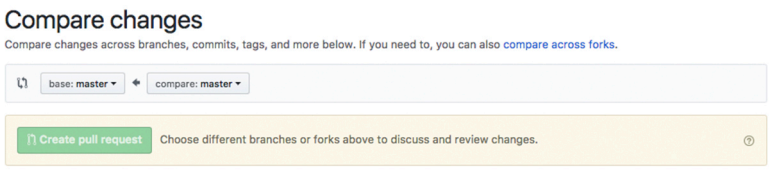
2) In the “Branch” menu, choose the branch that contains your commits.



3) To the right of the Branch menu, click New Pull Request



4) Use the base branch dropdown menu to select the branch you’d like to merge your changes into, then use the compare branch drowpdown menu to choose the topic branch you made your changes in

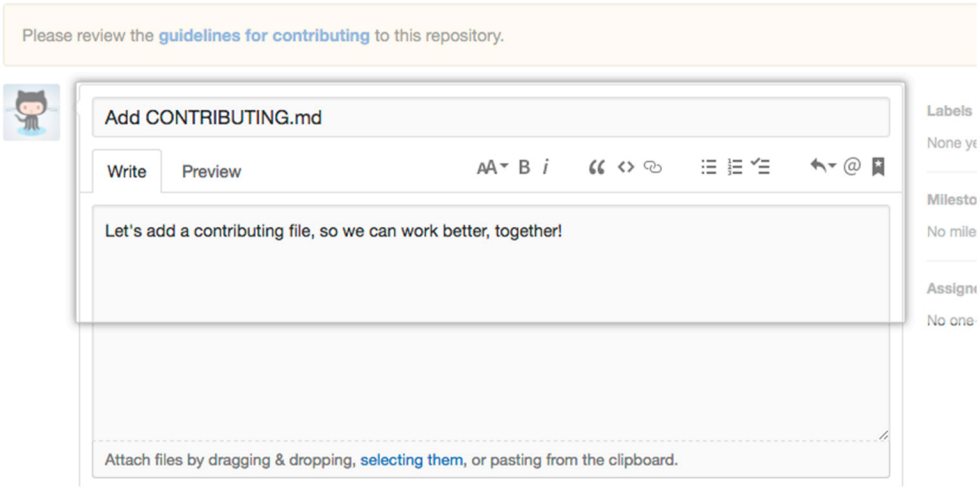


ZERO TO MASTERY

first contributions

a step by step guide

5) Type a title and description for your pull request.



6) Click create pull request

Congrats!

you've just made your first contribution

ZERO TO MASTERY

here are some resources!

- https://techcrunch.com/2012/07/14/what-exactly-is-github-anyway/
- https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/
- https://www.codefellows.org/blog/git-and-github-what-s-the-difference/
- https://www.udemy.com/the-complete-web-developer-in-2018/learn/v4/t/lecture/8725782?start=0
- https://rubygarage.org/blog/how-contribute-to-open-source-projects
- https://egghead.io/lessons/javascript-exploring-a-repository
- https://opensource.guide
- https://github.com/zero-to-mastery/start-here-guidelines/blob/master/Get%20Started.md
- https://help.github.com/articles/set-up-git/
- https://github.com/Roshanjossey/first-contributions
- https://opensource.guide/how-to-contribute/#how-to-submit-a-contribution

ZERO TO MASTERY

the complete web developer course
created by Andrei Neagoie

this guide was made with love by Wan Raitelli
@wanraitelli